intel®

AP-369

APPLICATION NOTE

# LAN557 Hardware/Software Interface Definition

August 1996 Order Number: 644863-002

# 1.0. INTRODUCTION

This document contains a hardware/software interface definition for the Intel 82557 10/100 Mbps LAN Controller. Hardware implemented to be compatible with this interface can work with software developed by Intel and other NOS vendors that conform to this interface definition. This definition is intended to facilitate designs, regardless of whether they are motherboard or adapter card implementations. For more information on the 82557, please refer to the *Intel 82557 User Manual* and *Datasheet*. Contact your local sales office to obtain the latest specifications.

*WARNING: Because it is impossible to predict all possible implementations, Intel cannot guarantee the functionality of Intel-written software when used on hardware designs that deviate from the definitions in this document.*

# 2.0. CONFIGURATION TECHNIQUES

## 2.1. Hardware Parameters

### 2.1.1. OLDER CONFIGURATION TECHNIQUES

LAN drivers usually support more than one hardware configuration. The parameters typically varied are: locations of the base address for I/O and memory decode, memory size, and selection of DMA and interrupt lines. On adapters, this configuration change has been commonly done through the use of jumpers and switches. On the motherboard, where it may be desirable or impossible to include switches, hardware configuration values may be either fixed or will utilize software (jumperless) configuration functions.

### 2.1.2. PCI: THE NEW CONFIGURATION TECHNIQUE

The 82557, as a PCI (Peripheral Component Interconnect) LAN controller, contains an integrated 32-bit PCI Bus Master interface. This PCI bus interface enables the 82557 to interact with the host system via the PCI bus. One of the main features of PCI is its support of Auto Configuration, i.e. automatic peripheral detection and configuration.

When a PC is first powered on, the configuration software must scan the PCI bus, or buses, to determine which PCI devices exist and what configuration requirements they have, build a consistent address map, and determine if a device has an expansion ROM. After determining an existence of the device, the configuration software (usually stored in a PCI compliant BIOS) then accesses the device's configuration registers to determine and allocate the blocks of memory and/or I/O space the device requires. Every device will have its own individual memory or I/O range which is guaranteed to be different than the other device residing on the PCI bus. In addition, the configuration software chooses how to distribute interrupt requests issued by various devices. Refer to the *PCI Bus Interface Specification* (currently rev. 2.1) for more details specific to PCI. Refer to the *82557 Datasheet* for more information on PCI configuration registers unique to the 82557 and the base address registers used to request memory mapped and I/O mapped resources.

## 2.2. Software Parameters

In addition to be able to modify the hardware parameters, the network driver itself must also be configurable. Drivers for Novell NetWare Version 3.X and beyond use the configuration information from command line parameters. NetWare ODI and NDIS drivers use configuration information from an ASCII text file that is separate from the driver.

# 3.0. DESIGN ISSUES

This specification is based on a number of possible design implementations. Implementations that deviate from these options may be possible, but Intel cannot guarantee that all drivers will work on that hardware.

**Option 1:** The design includes an EEPROM. It is connected directly to the 82557 to store hardware configuration parameters (i.e. node address, PWA number). Example implementations include an adapter or motherboard system.

**Option 2:** The design includes Flash. It is connected directly to the 82557 to store hardware configuration parameters (i.e. node address, PWA and/or RPL code).

Intel's LAN drivers use Option 1 to retrieve configuration information. Although they do not use Option 2, the 82557 contains a glueless interface to a FLASH and can be programmed to address up to 1 Mbyte of Flash. Refer to the *82557 User Manual* for more details on accessing Flash.

## 3.1. INDIVIDUAL ADDRESS STORAGE

Every 802.3 node has a unique 48-bit address assigned by the IEEE. It is called the MAC individual address because it is used by the MAC sub-layer. This 48-bit address must be stored in a nonvolatile form, such as in the EEPROM or Flash EPROM. Table 1 contains a list of recommended PROMs which may be used for storing the address.

**Table 1. Suggest PROMs**

| Type | Part Number | Speed Requirement |
|---|---|---|
| Serial EEPROM | Hyundai HY93C46 or Equivalent | 1us |
| Flash EPROM | 28F010 | 250ns |

## 3.2. FORMAT OF THE EEPROM

The format for the EEPROM is listed in Table 2. The EEPROM should be loaded with the default values before distributed to the end user. Note: The EEPROM uses a 16-bit word format rather than an 8-bit byte format.

**Table 2. 82557 EEPROM Format (100BASE-TX design)**

| Word Address | Fixed/ Variable | Description High Byte | Description Low Byte | Default Value (hex) |
|---|---|---|---|---|
| 0 | Variable | IA Byte 2 | IA Byte 1 | AA00 |
| 1 | Variable | IA Byte 4 | IA Byte 3 | IA(4)00 |
| 2 | Variable | IA Byte 6 | IA Byte 5 | IA(6,5) |
| 3 | Fixed | Compatibility_1 | Compatibility_0 | 0000 |
| 4 | Fixed | Reserved | Reserved | 0000 |
| 5 | Fixed | Controller Type | Connectors | 0101 |
| 6 | Fixed | Primary PHY Record, high byte | Primary PHY Record, low byte | 4401 |
| 7 | Fixed | Secondary PHY Record, high byte | Secondary PHY Record, low byte | 0000 |
| 8 | Fixed | PWA, byte 1 | PWA, byte 2 | 3525 |
| 9 | Fixed | PWA, byte 3 | PWA, byte 4 | 0903 |
| A | Fixed | RESERVED | RESERVED | 0000 |
| B | Fixed | Subsytem_ID, high byte | Subsytem_ID, low byte | 0000 |
| C | Fixed | Subsytem_Vendor, high byte | Subsytem_Vendor, low byte | 0000 |
| D-2F | Fixed | RESERVED | RESERVED | 0000 |
| 30-31 | Fixed | RPL Config | RPL Config | 0000 |
| 32-3E | Fixed | RESERVED | RESERVED | 0000 |
| 3F | Variable | Checksum, high byte | Checksum, low byte | Checksum of words 0-3E |

Table 3. 82557 EEPROM Format (100BASE-T4 design)

| Word Address | Fixed/ Variable | Description High Byte | Description Low Byte | Default Value (hex) |
|---|---|---|---|---|
| 0 | Variable | IA Byte 2 | IA Byte 1 | AA00 |
| 1 | Variable | IA Byte 4 | IA Byte 3 | IA(4)00 |
| 2 | Variable | IA Byte 6 | IA Byte`5 | IA(6,5) |
| 3 | Fixed | Compatibility_1 | Compatibility_0 | 0000 |
| 4 | Fixed | Reserved | Reserved | 0000 |
| 5 | Fixed | Controller Type | Connectors | 0101 |
| 6 | Fixed | Primary PHY Record, high byte | Primary PHY Record, low byte | 4101 |
| 7 | Fixed | Secondary PHY Record, high byte | Secondary PHY Record, low byte | 0000 |
| 8 | Fixed | PWA, byte 1 | PWA, byte 2 | 3524 |
| 9 | Fixed | PWA, byte 3 | PWA, byte 4 | 3303 |
| A | Fixed | RESERVED | RESERVED | 0000 |
| B | Fixed | Subsytem_ID, high byte | Subsytem_ID, low byte | 0000 |
| C | Fixed | Subsytem_Vendor, high byte | Subsytem_Vendor, low byte | 0000 |
| D-2F | Fixed | RESERVED | RESERVED | 0000 |
| 30-31 | Fixed | RPL Config | RPL Config | 0000 |
| 32-3E | Fixed | RESERVED | RESERVED | 0000 |
| 3F | Variable | Checksum, high byte | Checksum, low byte | Checksum of words 0-3E |

## 3.3. 100Base-TX / 100Base-T4 EEPROM Word Descriptions

### 3.3.1. ETHERNET ADDRESS FORMAT

The Ethernet Individual Address (IA) is a 6 byte field that must be unique for each adapter card, and thus unique for each copy of the EEPROM image.

#### 3.3.1.1. Controller Type

This byte wide field indicates which 82557 family controller is installed. All supported devices will share the same PCI Vendor and Product IDs (8086/1229).

#### 3.3.1.2. Connectors

This byte-wide field indicates which connector types are physically preset on the adapter product:

| D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
|---|---|---|---|---|---|---|---|
| RFU | RFU | RFU | RFU | MII | AUI | BNC | RJ-45 |

RJ-45: Supports 10BASE-T and 100BASE-T. Refer to the MII Status Register to determine which, if applicable, 100BASE-T technology is supported.

BNC: Supports 10BASE-2.

AUI: Supports 10BASE-5 Attachment Unit Interface.

MII: Supports 10BASE-T and/or 100BASE-T via an external transceiver. Refer to the MII Status Register to determine which, if applicable, 100BASE-T technology is provided. The presence of a PHY device on the MII connector is optional. However, if an external PHY device is attached via the MII connector, its PHY address must be 0.

RFU: Reserved for future use.

#### 3.3.1.3. PHY Device Record (Primary/Secondary)

The PHY Device Record allows the 82557 to work with a variety of PHYs using a single 82557 driver. Specifically, it provides the driver a listing of PHYs in order for the driver to identify the specific PHY used. Two PHY Device Records are provided to accommodate combo (multiple connector) products. Combo products with more than two PHYs are beyond the scope of this specification and are not supported with 82557 drivers at this time.

| High Byte | | | Low Byte |
|---|---|---|---|
| Bit 15 | Bit 14 | Bits 13-8 | Bits 7-0 |
| 10Mbps only | VSCR | PHYDevice | PHY Address |

- 10Mbps only: This bit will be set to 1 if the PHY is a 10 Mbps only device such as the Intel 82503 or Seeq 82C24. Implicit is the assumption that the PHY will use a serial interface (not MII), does not support the MII management interface, and will require the 82557 to be configured for the 82503 interface. The value in this field is not valid if the PHY device field is equal to 0.
- VSCR (Vendor Specific Code Required): This bit will be set to 1 if the PHY device requires special programming (either setting and/or reporting the correct operating mode) via vendor specific registers from the MII management interface. If this bit is 0 and the device is 100 Mbps/MII capable, then all status and control is achieved through the standard MII Register set defined by the 802.3u specification. If this bit is 0, and the device is 10 Mbps capable only, then media type selection will be performed automatically by the hardware. When this bit is 0, the device driver may use its standard configuration code path, even if the PHY Device code (or PHY identifier found via MII register) is not recognized. This mechanism allows new PHY devices to be designed onto the hardware allowing software compatibility with known devices. The value in this field is not valid if the PHY device field is equal to 0.
- PHY Device: This field contains an arbitrarily assigned code which uniquely identifies any device which may be used with the 82557.

| PHY Device Code | Vendor Product |
|---|---|
| 0 | No PHY device installed |
| 1 | Intel 82553 (PHY 100) A or B step |
| 2 | Intel 82553 (PHY 100) C step |
| 3 | Intel 82503 10Mps |
| 4 | National DP83840 100BASE-TX, C step |
| 5 | Seeq 80C240 - 100BASE-T4 |
| 6 | Seeq 80C24 - 10Mps |
| 7 | Intel 100Base-TX PHY |
| 8 | Microlinear 10Mbps |
| 9 | Level One 10Mbps |
| A | National Semiconductor DP83840A |
| 7-63 | Reserved for future use |

**NOTES:**
Currently, Intel 82557 drivers are not guaranteed to work with PHYs not listed in the table above. Please refer to any subsequent updates in this specification for further PHY support in Intel drivers.

- PHY Address: This field indicates the PHY address for the given device. The PHY Address is only relevant for devices which support the MII management interface. The value in this field is not valid if the PHY device is equal to 0 of if the device is marked as 10 Mbps only.

#### 3.3.1.4. PWA Number

The PWA (printed wiring assembly or printed board assembly) number is stored in a four-byte field and is unique to the vendor.

#### 3.3.1.5. RPL Configuration

This field provides 4 bytes of configuration information for the Remote Program Load application (RPL). This field does not require factory programming. It will be initialized and used only by the LanWorks* RPL application.

#### 3.3.1.6. Checksum Word

The checksum word (3Fh) should be calculated such that after adding all the words (00h-3Fh), including the Checksum word itself, the sum should be BABAh. The initial value in the summing register should be 0000h and the carry bit should be ignored after each addition.

## 4.0. SOFTWARE PARAMETERS: CUSTOM KEYWORDS FOR DRIVERS

The following is a recommended list of user supplied CUSTOM keywords. If specified, each of these keywords will override all other system defaults.

CUSTOM keywords will be divided into 2 categories, hidden and user-definable.

### 4.1. Hidden Keywords

These keywords should not be documented for the general user. They should be supported in software and made available to users as required.

Table 4. Hidden Keywords

| Parameter Name | Description | Default Value | Range Of Acceptable Values |
|---|---|---|---|
| TXFIFO | Threshold in DWORDs (bytes) in the 82557 internal 64-byte transmit FIFO. Only configurable at INIT time. | 8<br>(8 DWORDs or 32 bytes) | 0 To 0Fh Hexadecimal |
| RXFIFO | Threshold in DWORDs (bytes) in the 82557 internal 64-byte receive FIFO. Only configurable at INIT time. | 8<br>(8 DWORDs or 32 bytes) | 0 To 0Fh Hexadecimal |
| TXDMACOUNT | Setting to indicate the maximum number of TX DMA PCI cycles that will be completed after internal arbitration. A default value of 0 means that the TX DMA cannot be preempted. | 0<br>(No preemption) | 0 To 127 |
| RXDMACOUNT | Setting to indicate the maximum number of RX DMA PCI cycles that will be completed after internal arbitration. A default value of 0 means that the RX DMA cannot be preempted. | 0<br>(No preemption) | 0 To 127 |
| CONGENB | Enables congestion control in National TX PHY | 0<br>(No congestion control) | 0 Or 1 |

## 4.2. User-Definable Keywords

These CUSTOM keywords will be documented and can be changed or reconfigured by the general user. Users will be warned about their misuse, as these parameters will override all defaults.

**Table 5. User-Definable Keywords**

| Parameter Name | Description | Default Value | Range Of Acceptable Values |
|---|---|---|---|
| SPEED | Forces the line speed. Auto speed negotiation will be disabled in the PHY. Software will validate and use the FORCEFDX option only if SPEED is valid and specified as an override. | None<br>Auto speed negotiation will be enabled. | 10 Or 100 Decimal |
| IOMAPMODE | Parameter to force the use of I/O mode for CSR accesses. Software supporting memory and I/O modes can default to memory-mapped mode. | 0 (Disabled) | 0 Or 1 |
| FORCEFDUPLEX | Forces the 82557 into Full or Half Duplex mode (1 = half, 2 = full). This option is valid only if SPEED is specified and is valid. | 0 (Auto) | 0 - 2 |
| PHYADDRESS | Specifies the address for logical PHY 1. PHY 0 can only be at address 0. But PHY 1 can be at any address in the range 1 - 31 inclusive. PHY address 32 will be interpreted as 82503. | 1 | 0 - 32 |

## 5.0. ADDITIONAL INFORMATION

For additional literature contact your local Intel sales office or contact the Intel Literature Center by calling 1-800-548-4725. If you need design information, contact your local Intel Field Applications Engineer.